

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Агафонов Александр Викторович

Должность: директор филиала

Дата подписания: 18.06.2026 09:51:35

Университет: Московский политехнический университет

2539477a8ecf706dc9cff164bc411eb6d3c4ab06

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**  
**ВЫСШЕГО ОБРАЗОВАНИЯ «МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**  
**ЧЕБОКСАРСКИЙ ИНСТИТУТ (ФИЛИАЛ) МОСКОВСКОГО ПОЛИТЕХНИЧЕСКОГО УНИВЕРСИТЕТА**

## **Кафедра Информационных технологий и систем управления**



# **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

## **«Структуры и алгоритмы обработки данных»**

(наименование дисциплины)

Направление подготовки	<b>09.03.01 «Информатика и вычислительная техника»</b> (код и наименование направления подготовки)
Направленность (профиль) подготовки	<b>«Программное обеспечение вычислительной техники и автоматизированных систем»</b> (наименование профиля подготовки)
Квалификация выпускника	<b>бакалавр</b>
Форма обучения	<b>очная, заочная</b>
Год начала обучения	<b>2026</b>

Чебоксары, 2026

Рабочая программа дисциплины разработана в соответствии с:

- Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.01 – Информатика и вычислительная техника, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 929 от 19 сентября 2017 г. зарегистрированный в Минюсте 10 октября 2017 года, рег. номер 48489 (далее – ФГОС ВО);
- учебным планом (очной, заочной форм обучения) по направлению подготовки 09.03.01 «Информатика и вычислительная техника».

Рабочая программ дисциплины включает в себя оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине (п.6 Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины).

Автор Пикина Наталия Евгеньевна, кандидат педагогических наук, доцент кафедры информационных технологий и систем управления

*(указать ФИО, ученую степень, ученое звание или должность)*

Программа одобрена на заседании кафедры Информационных технологий и систем управления (протокол № 9 от 22.05.2026 г.).

# **1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы (Цели освоения дисциплины)**

1.1. *Целями* освоения дисциплины «Структуры и алгоритмы обработки данных» являются:

изучение применяемых в программировании структур данных, их спецификаций и реализаций в различных классах задач, изучение алгоритмов обработки данных, взаимосвязи алгоритмов и структур, получение и развитие навыков анализа алгоритмов, умения делать обоснованный выбор в пользу той или иной структуры данных при решении различных классов задач, умения применять алгоритмы на практике.

Для достижения целей дисциплины необходимо решить следующую *основную задачу* – сформировать у обучающихся теоретические знания и практические навыки, необходимые для:

формирования базовые теоретические понятия, лежащие в основе процесса разработки алгоритмов и структур данных;

заложения в основу конструирования и использования сложных (динамических) структур данных модель (парадигму) абстрактного типа данных (спецификация + представление + реализация);

формирования представления и знания об основных классах алгоритмов (исчерпывающий поиск, быстрый поиск, сортировки, алгоритмы на графах и т.п.), используемых в них структурах данных и общих схемах решения задач на их основе;

реализации типовых алгоритмов и структур данных и их модификаций на выбранном рабочем языке программирования;

формирования представления и знания об анализе сложности алгоритмов и программ.

1.2. Области профессиональной деятельности и (или) сферы профессиональной деятельности, в которых выпускники, освоившие программу, могут осуществлять профессиональную деятельность:

*Об Связь, информационные и коммуникационные технологии (в сфере исследования, разработки, внедрения и сопровождения информационных технологий и систем).*

1.3. К основным задачам изучения дисциплины относится подготовка обучающихся к выполнению трудовых функций в соответствии с профессиональными стандартами:

Код и наименование профессионального стандарта	Обобщенные трудовые функции			Трудовые функции		
	код	наименование	уровень квалификации	наименование	код	уровень (подуровень) квалификации
06.001 Программист Профессиональный стандарт «Программист», утвержденный приказом Министерством труда и социальной защиты Российской Федерации от 18.11.2013 № 679н (зарегистрирован в Министерстве труда и социальной защиты Российской Федерации от 18 декабря 2013 г. №30635)	D	Разработка требований и проектирование программного обеспечения	6	Анализ требований к программному обеспечению	D/01.6	6
			6	Разработка технических спецификаций на программные компоненты и их взаимодействие	D/02.6	
			6	Проектирование программного обеспечения	D/03.6	
06.028 Системный программист Профессиональный стандарт «Системный программист», утв. Приказом Министерством труда и социальной защиты РФ от 29 сентября 2020 года N 678н	A	Разработка компонентов системных программных продуктов	6	Разработка драйверов устройств	A/01.6	6
			6	Разработка компиляторов, загрузчиков, сборщиков	A/02.6	6
				Разработка системных утилит	A/03.6	6
				Создание инструментальных средств программирования	A/04.6	6
06.015 Специалист по информационным системам Профессиональный стандарт «Специалист по информационным системам», утв. приказом Министерством труда и социальной защиты РФ 18 ноября 2014 г. №896н	C	Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы	6	Разработка модели бизнес-процессов заказчика	C/08.6	6
				Выявление требований к ИС	C/11.6	6
				Анализ требований	C/12.6	6
				Разработка архитектуры ИС	C/14.6	6
				Проектирование и дизайн ИС	C/16.6	6
				Разработка баз данных ИС	C/17.6	6
				Организационное	C/18.6	6

Код и наименование профессионального стандарта	Обобщенные трудовые функции			Трудовые функции		
	код	наименование	уровень квалификации	наименование	код	уровень (подуровень) квалификации
				и технологическое обеспечение кодирования на языках программирования		
				Организационное и технологическое обеспечение модульного тестирования ИС (верификации)	C/19.6	6
				Создание пользовательской документации к ИС	C/22.6	6
				Организация репозитория хранения данных о создании (модификации) и вводе ИС в эксплуатацию	C/40.6	6

#### 1.4. Компетенции обучающегося, формируемые в результате освоения дисциплины

Наименование категории (группы) компетенций	Код и наименование компетенций	Код и наименование индикатора достижения компетенции	Перечень планируемых результатов обучения
Разработка, отладка, модификация и поддержка системного программного обеспечения	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств	<p><i>на уровне знаний:</i>  знать методы и средства разработки компонентов аппаратно-программных комплексов и баз данных;</p> <p><i>на уровне умений:</i>  уметь применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных</p> <p><i>на уровне навыков:</i>  владеть методами и средствами разработки компонентов аппаратно-</p>

			программных комплексов и баз данных
		ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики	<p><i>на уровне знаний:</i> знать стандарты, методические и нормативные материалы, определяющие проектирование и разработку компонентов аппаратно-программных комплексов и баз данных;</p> <p><i>на уровне умений:</i> уметь применять современные программно-методические комплексы автоматизированного проектирования объектов профессиональной деятельности;</p> <p><i>на уровне навыков:</i> методами организации процесса разработки компонентов аппаратно-программных комплексов и баз данных.</p>
		ПК-3.3 Разрабатывает системные утилиты	<p><i>на уровне знаний:</i> знать модели, методы и формы организации процесса разработки компонентов аппаратно-программных комплексов и баз данных;</p> <p><i>на уровне умений:</i> уметь применять методы и средства обеспечения информационной безопасности разрабатываемых компонентов</p> <p><i>на уровне навыков:</i> владеть методами и средствами обеспечения информационной безопасности разрабатываемых компонентов</p>

		ПК-3.4. Создает инструментальные средства программирования	<p><i>на уровне знаний:</i>  знать стандарты, руководящих документов и норм, регламентирующих создание и сопровождение структур обработки данных и баз данных.</p> <p><i>на уровне умений:</i>  уметь эффективно применять современные инструменты и методы автоматизации проектирования структур обработки данных и баз данных.</p> <p><i>на уровне навыков:</i>  владеть приёмами и технологиями, обеспечивающими организацию эффективного процесса разработки компонент баз данных и структур обработки данных. <i>Начало формы</i></p>
--	--	--	--

## 2. Место дисциплины в структуре ОПОП

Дисциплина Б1.Д(М).В.3 «Структуры и алгоритмы обработки данных» в рамках части, формируемой участниками образовательных отношений блока «Элективные дисциплины (модули)» программы бакалавриата..

Дисциплина «Структуры и алгоритмы обработки данных» преподаётся обучающимся по очной форме обучения – в 4-м семестре, по заочной форме – в 5-м семестре.

Дисциплина «Структуры и алгоритмы обработки данных» является промежуточным этапом формирования компетенций ПК-3 в процессе освоения ОПОП.

Дисциплина «Структуры и алгоритмы обработки данных» основывается на знаниях, умениях и навыках, приобретенных при изучении дисциплин Перспективы развития информатики и вычислительной техники, Введение в информатику, Учебная практика: ознакомительная практика, Объектно-ориентированное программирование, Программирование для мобильных устройств и является предшествующей для изучения дисциплин Криптографические методы защиты информации, Защита информации, Учебная практика: научно-исследовательская работа (получение первичных навыков научно-исследовательской работы), Учебная практика: технологическая (проектно-технологическая) практика, Микропроцессорные устройства систем управления, Интернет-технологии, Интернет-программирование, Функциональное и логическое программирование, Системное программирование, Государственная итоговая аттестация: подготовка к сдаче и

сдача государственного экзамена, Государственная итоговая аттестация: выполнение и защита выпускной квалификационной работы.

Формой промежуточной аттестации знаний обучаемых по очной форме обучения является курсовая работа и экзамен в 4-м семестре, по заочной форме курсовая работа и экзамен в 5-м семестре.

### 3. Объем дисциплины

очная форма обучения:

Вид учебной работы по дисциплине	Всего в з.е. и часах	Семестр 4 в часах
<b>Общая трудоёмкость дисциплины</b>	<b>5 з.е. - 180 ак.час</b>	<b>180 ак.час</b>
<b>Контактная работа - Аудиторные занятия</b>	<b>75</b>	<b>75</b>
<i>Лекции</i>	36	36
<i>Лабораторные занятия</i>	36	36
<i>Семинары, практические занятия</i>	-	-
<i>Консультация</i>	1	1
<b>Самостоятельная работа</b>	<b>69</b>	<b>69</b>
<b>Курсовая работа (курсовой проект)</b>	2	2
<b>Вид промежуточной аттестации</b>	Экзамен – 36 часов	Экзамен – 36 часов

заочная форма обучения:

Вид учебной работы по дисциплине	Всего в з.е. и часах	Семестр 5 в часах
<b>Общая трудоёмкость дисциплины</b>	<b>5 з.е. - 180 ак.час</b>	<b>180 ак.час</b>
<b>Контактная работа - Аудиторные занятия</b>	<b>19</b>	<b>19</b>
<i>Лекции</i>	8	8
<i>Лабораторные занятия</i>	8	8
<i>Семинары, практические занятия</i>	-	-
<i>Консультация</i>	1	1
<b>Самостоятельная работа</b>	<b>152</b>	<b>152</b>
<b>Курсовая работа (курсовой проект)</b>	2	2
<b>Вид промежуточной аттестации</b>	Экзамен – 9 часов	Экзамен – 9 часов

**4. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий**

#### 4.1. Учебно-тематический план

Очная форма обучения

Тема (раздел)	Распределение часов			Самосто я- тельная работа	Формируемые компетенции (код)
	Лекции	Лабораторн ые занятия	Практическ ие занятия		
Тема 1. Введение в дисциплину. Типы и структуры данных.	6	6	-	10	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 2. Типовые	6	6	-	11	ПК-3.1., ПК-3.2.,

алгоритмы обработки данных					ПК-3.3., ПК-3.4.
Тема 3. Статические и динамические агрегатные структуры данных	6	6	-	12	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 4. Структура данных дерево и алгоритмы на деревьях	6	6	-	12	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 5. Структура данных граф и алгоритмы на графах	6	6	-	12	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 6. Вычислительные и сложные алгоритмы	6	6	-	12	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Консультация	1			-	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Курсовая работа (курсовой проект)	2				ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Контроль (экзамен)	36				ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
<b>ИТОГО</b>	<b>75</b>			<b>69</b>	

### Заочная форма обучения

Тема (раздел)	Распределение часов			Самостоятельная работа	Формируемые компетенции (код)
	Лекции	Лабораторные занятия	Практические занятия		
Тема 1. Введение в дисциплину. Типы и структуры данных.	2	-	-	24	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 2. Типовые алгоритмы обработки данных	2	-	-	24	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 3. Статические и динамические агрегатные структуры данных	2	2	-	26	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 4. Структура данных дерево и алгоритмы на деревьях	2	2	-	26	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 5. Структура данных граф и алгоритмы на графах	-	2	-	26	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Тема 6. Вычислительные и сложные алгоритмы	-	2	-	26	ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Консультация	1			-	ПК-3.1., ПК-3.2., ПК-3.3.,

			ПК-3.4.
Курсовая работа (курсовой проект)	2		ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
Контроль (экзамен)	9		ПК-3.1., ПК-3.2., ПК-3.3., ПК-3.4.
<b>ИТОГО</b>	<b>19</b>	<b>152</b>	

## 4.2. Содержание дисциплины

### Тема 1. Введение в дисциплину. Типы и структуры данных.

Понятие структуры данных. Классификация типов данных: простые (числа, символы, логические), составные (массивы, записи, множества).

Абстрактные типы данных (АТД): стек, очередь, список, множество, словарь.

Принципы организации и представления данных в памяти. Отличие логической и физической структур.

Понятие эффективности алгоритмов: время выполнения, используемая память, сложность.

Роль структур данных в программировании и инженерных задачах. Обзор применимости различных структур в типовых задачах.

Связь между структурами данных и алгоритмами их обработки.

### Тема 2. Типовые алгоритмы обработки данных

Обзор базовых алгоритмов: линейный поиск, бинарный поиск, подсчёт, фильтрация, свёртка.

Алгоритмы сортировки: пузырьковая, выбором, вставками, быстрая сортировка (QuickSort), сортировка слиянием (MergeSort), пирамидальная сортировка (HeapSort).

Сравнение алгоритмов по сложности: наилучший, худший и средний случай.

Рекурсия: принципы работы, стек вызовов, хвостовая рекурсия. Примеры рекурсивных алгоритмов.

Итерация и рекурсия: выбор подхода в зависимости от задачи.

Анализ производительности алгоритмов с использованием нотации «О-большое» (Big-O).

### Тема 3. Статические и динамические агрегатные структуры данных

Массивы (одномерные и многомерные): организация, доступ к элементам, преимущества и ограничения.

Структуры (records), объединения (unions), перечисления.

Связные списки: односвязные, двусвязные, кольцевые. Операции вставки, удаления, поиска.

Динамическое выделение и освобождение памяти. Указатели и ссылки.

Очереди и стеки: принципы работы, реализация на массивах и списках.  
Кольцевые буферы, дека (двусторонняя очередь). Практические применения в алгоритмах и ОС.

#### **Тема 4. Структура данных дерево и алгоритмы на деревьях**

Деревья: определения, терминология (корень, листья, уровень, высота).  
Представление в памяти.

Бинарные деревья, бинарные деревья поиска (BST). Алгоритмы вставки, удаления, поиска.

Обходы дерева: прямой (pre-order), симметричный (in-order), обратный (post-order), в ширину (BFS).

Сбалансированные деревья: AVL-деревья, красно-чёрные деревья.  
Поддержание баланса.

Куча (heap) и её применение в алгоритмах сортировки и приоритетных очередях.

Обработка и анализ иерархических данных. Использование деревьев в компиляторах, базах данных и файловых системах.

#### **Тема 5. Структура данных граф и алгоритмы на графах**

Графы: ориентированные и неориентированные, взвешенные и невзвешенные. Представление: матрица смежности, списки смежности.

Обход графов: в глубину (DFS), в ширину (BFS). Применение для поиска и анализа связности.

Алгоритмы поиска кратчайших путей: Дейкстры, Беллмана-Форда, Флойда-Уоршелла.

Алгоритмы построения остовных деревьев: Прима, Крускала.

Поиск компонент связности, топологическая сортировка, алгоритмы обнаружения циклов.

Применение графов: маршрутизация, социальные сети, анализ зависимостей, графы переходов состояний.

#### **Тема 6. Вычислительные и сложные алгоритмы**

Классы сложности задач: P, NP, NP-полные и NP-трудные задачи.  
Примеры и значимость в практике.

Жадные алгоритмы: принципы построения, условия применимости, примеры (задача о рюкзаке, покрытие множества).

Методы динамического программирования: разбиение задачи, таблицы, примеры (поиск наибольшей общей подпоследовательности, задача о размене).

Разделяй и властвуй: концепция, примеры (быстрая сортировка, бинарный поиск, умножение больших чисел).

Алгоритмы на строках: поиск подстроки (алгоритм Кнута-Морриса-Пратта, Бойера-Мура), хеширование.

Алгоритмы с использованием хеш-таблиц и множеств. Эффективная организация доступа к данным.

### **5. Учебно-методическое обеспечение самостоятельной работы студентов**

Самостоятельная работа проводится с целью: систематизации и закрепления полученных теоретических знаний и практических умений обучающихся; углубления и расширения теоретических знаний студентов; формирования умений использовать нормативную, правовую, справочную документацию, учебную и специальную литературу; развития познавательных способностей и активности обучающихся: творческой инициативы, самостоятельности, ответственности, организованности; формирование самостоятельности мышления, способностей к саморазвитию, совершенствованию и самоорганизации; формирования профессиональных компетенций; развитию исследовательских умений студентов.

Формы и виды самостоятельной работы студентов: чтение основной и дополнительной литературы – самостоятельное изучение материала по рекомендуемым литературным источникам; работа с библиотечным каталогом, самостоятельный подбор необходимой литературы; работа со словарем, справочником; поиск необходимой информации в сети Интернет; конспектирование источников; реферирование источников; составление аннотаций к прочитанным литературным источникам; составление рецензий и отзывов на прочитанный материал; составление обзора публикаций по теме; составление и разработка терминологического словаря; составление хронологической таблицы; составление библиографии (библиографической картотеки); подготовка к различным формам текущей и промежуточной аттестации (к тестированию, контрольной работе, экзамену); выполнение домашних контрольных работ; самостоятельное выполнение практических заданий репродуктивного типа (ответы на вопросы, задачи, тесты; выполнение творческих заданий).

Технология организации самостоятельной работы обучающихся включает использование информационных и материально-технических ресурсов образовательного учреждения: библиотеку с читальным залом, компьютерные классы с возможностью работы в Интернет; аудитории (классы) для консультационной деятельности.

Перед выполнением обучающимися внеаудиторной самостоятельной работы преподаватель проводит консультирование по выполнению задания, который включает цель задания, его содержания, сроки выполнения, ориентировочный объем работы, основные требования к результатам работы, критерии оценки. Во время выполнения обучающимися внеаудиторной самостоятельной работы и при необходимости преподаватель может проводить индивидуальные и групповые консультации.

Контроль самостоятельной работы студентов предусматривает: соотнесение содержания контроля с целями обучения; объективность контроля; валидность контроля (соответствие предъявляемых заданий тому, что предполагается проверить); дифференциацию контрольно-измерительных материалов.

Формы контроля самостоятельной работы: просмотр и проверка выполнения самостоятельной работы преподавателем; организация самопроверки, взаимопроверки выполненного задания в группе; обсуждение

результатов выполненной работы на занятии; проведение письменного опроса; проведение устного опроса; организация и проведение индивидуального собеседования; организация и проведение собеседования с группой.

**Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы**

<b>Наименование тем (разделов) дисциплины</b>	<b>Перечень вопросов, отводимых на самостоятельное освоение</b>	<b>Формы внеаудиторной самостоятельной работы</b>
Тема 1. Введение в дисциплину. Типы и структуры данных.	<ol style="list-style-type: none"> <li>1. Основные понятия и задачи дисциплины.</li> <li>2. Абстрактные типы данных: стек, очередь, список.</li> <li>3. Классификация структур данных: линейные, иерархические, графовые.</li> <li>4. Примитивные типы данных и их представление в памяти.</li> <li>5. Различия между структурами данных и типами данных.</li> <li>6. Понятие сложности алгоритмов. Временная и пространственная сложность.</li> </ol>	<p>Решение задач на определение типа структуры данных для заданных условий.</p> <p>Составление сравнительной таблицы основных типов данных по критериям использования.</p>
Тема 2. Типовые алгоритмы обработки данных	<ol style="list-style-type: none"> <li>1. Алгоритмы поиска: линейный, бинарный, интерполяционный.</li> <li>2. Алгоритмы сортировки: пузырьком, вставками, выбором, быстрая сортировка, сортировка слиянием.</li> <li>3. Сравнение эффективности различных алгоритмов сортировки.</li> <li>4. Использование рекурсии в алгоритмах обработки данных.</li> <li>5. Алгоритмы фильтрации и агрегации данных.</li> <li>6. Понятие стабильности сортировки.</li> </ol>	<p>Реализация и тестирование алгоритмов сортировки на разных наборах данных.</p> <p>Сравнительный анализ сложности алгоритмов сортировки. Выполнение заданий по визуализации работы сортировок.</p>
Тема 3. Статические и динамические агрегатные структуры данных	<ol style="list-style-type: none"> <li>1. Статические структуры: массивы, фиксированные таблицы.</li> <li>2. Динамические структуры: связные списки (односвязные, двусвязные, циклические).</li> <li>3. Вставка, удаление и обход элементов в динамических структурах.</li> <li>4. Разница между стеком и очередью. Примеры применения.</li> <li>5. Реализация списков в языках программирования (C++, Python, Java).</li> <li>6. Применение агрегатных структур в решении прикладных задач.</li> </ol>	<p>Реализация динамических структур данных.</p> <p>Построение графического представления операций со списками.</p>
Тема 4. Структура данных дерево и алгоритмы на деревьях	<ol style="list-style-type: none"> <li>1. Понятие дерева и терминология: корень, листья, узлы, уровни.</li> <li>2. Бинарные деревья и их свойства.</li> <li>3. Алгоритмы обхода дерева: прямой, обратный, симметричный.</li> <li>4. Двоичные деревья поиска. Вставка и удаление элементов.</li> <li>5. AVL-деревья, красно-черные деревья: основные идеи и балансировка.</li> </ol>	<p>Разработка алгоритмов обхода дерева.</p> <p>Визуализация построения и балансировки дерева поиска. Анализ преимуществ сбалансированных деревьев.</p>

	6. Применение деревьев в системах хранения и поиска данных (например, XML, файловые системы).	
Тема 5. Структура данных граф и алгоритмы на графах	<ol style="list-style-type: none"> <li>1. Основные понятия: вершина, ребро, путь, цикл, связность.</li> <li>2. Представление графов: списки смежности, матрицы смежности и инцидентности.</li> <li>3. Алгоритмы обхода графа: в глубину (DFS), в ширину (BFS).</li> <li>4. Поиск кратчайших путей: алгоритм Дейкстры, Беллмана-Форда.</li> <li>5. Поиск минимального остовного дерева: алгоритмы Краскала и Прима.</li> <li>6. Примеры использования графов в задачах маршрутизации, анализа социальных сетей и др.</li> </ol>	<p>Реализация алгоритмов поиска в графе.          Моделирование реальной задачи (например, маршрутизация) с помощью графа.          Сравнение алгоритмов поиска кратчайшего пути по критериям времени выполнения.</p>
Тема 6. Вычислительные и сложные алгоритмы	<ol style="list-style-type: none"> <li>1. Понятие вычислимости и труднорешаемых задач.</li> <li>2. Алгоритмы "разделяй и властвуй", "жадные", "динамическое программирование".</li> <li>3. Классы сложности задач: P, NP, NP-полные задачи.</li> <li>4. Разработка эффективных алгоритмов для сложных структур.</li> <li>5. Примеры вычислительно сложных задач (задача коммивояжера, рюкзака и др.).</li> <li>6. Подходы к оптимизации и аппроксимации.</li> </ol>	<p>Решение задач с использованием динамического программирования.          Составление отчета по NP-полным задачам и возможным приближенным решениям. Программная реализация одного из сложных алгоритмов на примере.</p>

### Шкала оценивания

Шкала оценивания	Критерии оценивания
«Отлично»	Обучающийся глубоко и содержательно раскрывает тему самостоятельной работы, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер
«Хорошо»	Обучающийся в целом раскрывает тему самостоятельной работы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера
«Удовлетворительно»	Обучающийся в целом раскрывает тему самостоятельной работы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не владеет выбранной темой самостоятельной работы

**6. Оценочные материалы (фонд оценочных средств) для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины**

## 6.1. Паспорт фонда оценочных средств

№	Контролируемые разделы (темы) дисциплины	Код и наименование компетенции	Индикатор достижения компетенции	Наименование оценочного средства
1.	Тема 1. Введение в дисциплину. Типы и структуры данных. Основные понятия и определения	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4. Создает инструментальные средства программирования	Опрос, доклад, тест, реферат, курсовая работа, экзамен
2.	Тема 2. Типовые алгоритмы обработки данных	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4. Создает инструментальные средства программирования	Опрос, доклад, тест, реферат, курсовая работа, экзамен
3.	Тема 3. Статические и динамические агрегатные структуры данных	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4. Создает инструментальные средства программирования	Опрос, доклад, тест, реферат, курсовая работа, экзамен
4.	Тема 4. Структура данных дерево и алгоритмы на деревьях	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4. Создает инструментальные средства программирования	Опрос, доклад, тест, реферат, курсовая работа, экзамен
5.	Тема 5. Структура данных граф и алгоритмы на графах	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4. Создает	Опрос, доклад, тест, реферат, курсовая работа, экзамен

			инструментальные средства программирования	
6.	Тема 6. Вычислительные и сложные алгоритмы	ПК-3. Способен разрабатывать компоненты системных программных продуктов	ПК-3.1 Разрабатывает драйверы устройств ПК-3.2 Разрабатывает компиляторы, загрузчики, сборщики ПК-3.3 Разрабатывает системные утилиты ПК-3.4. Создает инструментальные средства программирования	Опрос, доклад, тест, реферат, курсовая работа, экзамен

**Этапы формирования компетенций в процессе освоения ОПОП** прямо связаны с местом дисциплин в образовательной программе. Каждый этап формирования компетенции, характеризуется определенными знаниями, умениями и навыками и (или) опытом профессиональной деятельности, которые оцениваются в процессе текущего контроля успеваемости, промежуточной аттестации по дисциплине (практике) и в процессе итоговой аттестации.

Дисциплина «Структуры и алгоритмы обработки данных» является промежуточным этапом комплекса дисциплин, в ходе изучения которых у студентов формируются компетенции ПК-3.

Формирования компетенции ПК-3 начинается с изучения дисциплин «Перспективы развития информатики и вычислительной техники», «Введение в информатику», Учебная практика (ознакомительная практика), «Объектно-ориентированное программирование», «Программирование для мобильных устройств».

Завершается работа по формированию у студентов указанных компетенций в ходе дисциплин «Криптографические методы защиты информации», «Защита информации», Учебная практика: научно-исследовательская работа (получение первичных навыков научно-исследовательской работы), Учебная практика: технологическая (проектно-технологическая) практика, «Микропроцессорные устройства систем управления», «Интернет-технологии», «Интернет-программирование», «Функциональное и логическое программирование», «Системное программирование».

Итоговая оценка сформированности компетенций ПК-3 определяется в период Государственная итоговая аттестация: подготовка к сдаче и сдача государственного экзамена, Государственная итоговая аттестация: выполнение и защита выпускной квалификационной работы.

**В процессе изучения дисциплины, компетенции также формируются поэтапно.**

Основными этапами формирования ПК-3 при изучении дисциплины Б1.Д(М).В.3 «Структуры и алгоритмы обработки данных» является последовательное изучение содержательно связанных между собой тем учебных

занятий. Изучение каждой темы предполагает овладение студентами необходимыми дескрипторами (составляющими) компетенций. Для оценки уровня сформированности компетенций в процессе изучения дисциплины предусмотрено проведение текущего контроля успеваемости по темам (разделам) дисциплины и промежуточной аттестации по дисциплине – курсовая работа и экзамен.

## **6.2. Контрольные задания и материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы**

### **6.2.1. Контрольные вопросы по темам (разделам) для опроса на занятиях**

Тема (раздел)	Вопросы
Тема 1. Введение в дисциплину. Типы и структуры данных. Основные понятия и определения	ПК-3 1. Введение в дисциплину. 2. Типы и структуры данных. 3. Основные понятия и определения 4. Архитектура баз данных 5. Понятие структуры данных: назначение и классификация. 6. Базовые типы данных: целые, вещественные, символы, логические. 7. Абстрактные типы данных (АТД): определение и примеры. 8. Отличия между типами и структурами данных. 9. Внутреннее и внешнее представление данных в памяти компьютера. 10. Связь между структурой данных и эффективностью алгоритма. 11. Основные принципы построения структур данных. 12. Примеры использования структур данных в прикладных задачах.
Тема 2. Типовые алгоритмы обработки данных	ПК-3 1. Типовые алгоритмы обработки данных 2. Прохождение пользовательского запроса 3. Пользователи банков данных 4. Классификация моделей данных 5. Классификация алгоритмов обработки данных. 6. Алгоритмы поиска: линейный и бинарный поиск. 7. Алгоритмы сортировки: пузырьковая, быстрая, сортировка слиянием. 8. Сравнение эффективности различных алгоритмов сортировки. 9. Алгоритмы вставки, удаления и обновления элементов в структуре. 10. Понятие сложности алгоритма: временная и пространственная. 11. Рекурсивные и итеративные алгоритмы: отличия и область применения. 12. Практические примеры применения типовых алгоритмов.

<p>Тема 3. Статические и динамические агрегатные структуры данных</p>	<p>ПК-3</p> <ol style="list-style-type: none"> <li>1. Статические агрегатные структуры данных</li> <li>2. Динамические агрегатные структуры данных</li> <li>3. Базовые понятия реляционной БД</li> <li>4. Концепция реляционной модели</li> <li>5. Массивы как пример статической структуры данных.</li> <li>6. Списки (односвязные, двусвязные): особенности реализации.</li> <li>7. Сравнение массивов и списков: преимущества и недостатки.</li> <li>8. Стек и очередь: принципы LIFO и FIFO.</li> <li>9. Реализация стека и очереди с использованием массива и списка.</li> <li>10. Кольцевая очередь: назначение и реализация.</li> <li>11. Примеры применения агрегатных структур в реальных задачах.</li> <li>12. Особенности динамического выделения и освобождения памяти.</li> </ol>
<p>Тема 4. Структура данных дерево и алгоритмы на деревьях</p>	<p>ПК-3</p> <ol style="list-style-type: none"> <li>1. Основные понятия реляционной алгебры</li> <li>2. Каскадное удаление и обновление данных</li> <li>3. Структура данных дерево</li> <li>4. Алгоритмы на деревьях</li> <li>5. Понятие дерева: узлы, корень, листья, уровень, высота.</li> <li>6. Двоичное дерево и его разновидности.</li> <li>7. Алгоритмы обхода дерева: прямой, симметричный, обратный.</li> <li>8. Бинарные деревья поиска: структура и применение.</li> <li>9. Самобалансирующиеся деревья (AVL, красно-черные): необходимость и принципы.</li> <li>10. Построение, вставка и удаление в деревьях.</li> <li>11. Использование деревьев в базах данных и индексировании.</li> <li>12. Деревья выражений и их применение в компиляторах.</li> </ol>
<p>Тема 5. Структура данных граф и алгоритмы на графах</p>	<p>ПК-3</p> <ol style="list-style-type: none"> <li>1. Структура данных граф</li> <li>2. Алгоритмы на графах</li> <li>3. Традиционные операции над множествами: объединение, пересечение, разность и декартово произведение.</li> <li>4. Специальные реляционные операции: выборка, проекция, соединение, деление.</li> <li>5. Представление графа: списки смежности и матрицы смежности.</li> <li>6. Направленные и ненаправленные графы: отличия и примеры.</li> <li>7. Взвешенные и невзвешенные графы.</li> <li>8. Поиск в глубину (DFS) и в ширину (BFS): принципы и применение.</li> <li>9. Алгоритм Дейкстры для поиска кратчайшего пути.</li> <li>10. Алгоритм Крускала и Прима для построения минимального остовного дерева.</li> <li>11. Поиск в графе циклов и компонентов связности.</li> <li>12. Примеры использования графов в сетевых и логистических задачах.</li> </ol>
<p>Тема 6. Вычислительные и сложные алгоритмы</p>	<p>ПК-3</p> <ol style="list-style-type: none"> <li>1. Вычислительные алгоритмы</li> <li>2. Сложные алгоритмы</li> <li>3. DDL операторы языка SQL</li> <li>4. DML операторы языка SQL</li> <li>5. DQL операторы языка SQL</li> <li>6. Понятие вычислительной сложности алгоритма.</li> </ol>

	<p>7. Классы сложности P, NP, NP-полные задачи.</p> <p>8. Асимптотическая оценка алгоритма: <math>O(n)</math>, <math>O(\log n)</math>, <math>O(n^2)</math> и др.</p> <p>9. Примеры задач с экспоненциальной и полиномиальной сложностью.</p> <p>10. Жадные алгоритмы: принципы работы и примеры.</p> <p>11. Динамическое программирование: суть метода и примеры.</p> <p>12. Разделяй и властвуй: структура и применение.</p> <p>13. Эвристики и приближённые алгоритмы для NP-полных задач.</p>
--	--

### Шкала оценивания ответов на вопросы

Шкала оценивания	Критерии оценивания
«Отлично»	Обучающийся глубоко и содержательно раскрывает ответ на каждый теоретический вопрос, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер.
«Хорошо»	Обучающийся в целом раскрывает теоретические вопросы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера.
«Удовлетворительно»	Обучающийся в целом раскрывает теоретические вопросы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не знает ответов на поставленные теоретические вопросы.

### 6.2.2. Темы для докладов

Тема (раздел)	Вопросы
Тема 1. Введение в дисциплину. Типы и структуры данных. Основные понятия и определения	<p>ПК-3</p> <p>1. Обзор типов данных и их важность в программировании.</p> <p>2. Статические и динамические структуры данных: различия и примеры.</p> <p>3. Абстрактные типы данных: определения и примеры.</p> <p>4. Основные операции с данными: вставка, удаление, поиск, сортировка.</p> <p>5. Массивы и их использование: преимущества и недостатки.</p> <p>6. Связанные списки: концепция, реализация и примеры.</p> <p>7. Стек и очередь: основные операции и применения.</p> <p>8. Хеш-таблицы: как они работают и где применяются.</p> <p>9. Алгоритмическая сложность операций с различными структурами данных.</p> <p>10. Роль структур данных в эффективности алгоритмов.</p>
Тема 2. Типовые алгоритмы обработки данных	<p>ПК-3</p> <p>1. Алгоритмы сортировки: пузырьковая сортировка, быстрая сортировка, сортировка слиянием.</p> <p>2. Поиск в массиве: линейный и бинарный поиск.</p> <p>3. Алгоритмы поиска в графах: поиск в глубину, поиск в ширину.</p> <p>4. Алгоритмы динамического программирования: принципы и примеры.</p> <p>5. Алгоритмы на строках: Кнута-Морриса-Пратта, Бойера-Мура.</p> <p>6. Применение жадных алгоритмов в задачах оптимизации.</p>

	<p>7. Алгоритмы на деревьях: обход в глубину и ширину, поиск на дереве.</p> <p>8. Алгоритмы нахождения кратчайших путей в графах: алгоритм Дейкстры, алгоритм Флойда.</p> <p>9. Применение хеш-функций в обработке данных.</p> <p>10. Сложность алгоритмов и её влияние на производительность.</p>
Тема 3. Статические и динамические агрегатные структуры данных	<p>ПК-3</p> <p>1. Статические структуры данных: массивы и их применения.</p> <p>2. Динамические структуры данных: связанный список и его разновидности.</p> <p>3. Агрегатные структуры данных: стек, очередь, приоритетная очередь.</p> <p>4. Бинарные деревья: структура и основные операции.</p> <p>5. Динамическая память и её управление в контексте структур данных.</p> <p>6. Адаптивные структуры данных: использование динамических структур для оптимизации операций.</p> <p>7. Статические и динамические хеш-таблицы: различия и применения.</p> <p>8. Балансировка деревьев и поддержка их динамичности.</p> <p>9. Преимущества и недостатки статических и динамических структур в зависимости от задачи.</p> <p>10. Применение агрегатных структур данных в реальных задачах.</p>
Тема 4. Структура данных дерево и алгоритмы на деревьях	<p>ПК-3</p> <p>1. Введение в структуры данных типа «дерево».</p> <p>2. Бинарные деревья: структура, операции и их сложность.</p> <p>3. Деревья поиска: балансировка и сортировка.</p> <p>4. Деревья отрезков и их использование в запросах к данным.</p> <p>5. Алгоритмы поиска и сортировки на бинарных деревьях.</p> <p>6. Деревья поиска с самобалансировкой: AVL-деревья, красно-черные деревья.</p> <p>7. Применение деревьев в вычислительных задачах и приложениях.</p> <p>8. Алгоритм обхода деревьев: рекурсивный и итеративный методы.</p> <p>9. Преобразование деревьев: преобразование в дерево поиска, в бинарное дерево.</p> <p>10. Проблемы балансировки деревьев и их решение.</p>
Тема 5. Структура данных граф и алгоритмы на графах	<p>ПК-3</p> <p>1. Основы теории графов: вершины, рёбра, типы графов (ориентированные, неориентированные).</p> <p>2. Матрицы смежности и списки смежности: представление графа в памяти.</p> <p>3. Алгоритм поиска в глубину и его применение в графах.</p> <p>4. Алгоритм поиска в ширину: суть, использование и примеры.</p> <p>5. Алгоритм Дейкстры для поиска кратчайшего пути.</p> <p>6. Применение алгоритмов поиска в графах в сетевых задачах.</p> <p>7. Топологическая сортировка ориентированных ациклических графов.</p> <p>8. Проблемы, решаемые с помощью графов: поиск пути, нахождение компонент связности.</p> <p>9. Алгоритмы поиска циклов в графах и их использование.</p> <p>10. Алгоритм Флойда для нахождения кратчайших путей в графах с несколькими источниками.</p>

<p>Тема 6. Вычислительные и сложные алгоритмы</p>	<p>ПК-3</p> <ol style="list-style-type: none"> <li>1. Введение в вычислительные алгоритмы и их сложность.</li> <li>2. Оценка сложности алгоритмов: большие <math>O</math>, <math>\Omega</math> и <math>\Theta</math>.</li> <li>3. Применение алгоритмов сортировки для улучшения производительности.</li> <li>4. Алгоритмы динамического программирования: решающее правило для сложных задач.</li> <li>5. Разделяй и властвуй: как делить задачу на подзадачи для улучшения вычислений.</li> <li>6. Алгоритмы поиска и сортировки на больших данных.</li> <li>7. Аппроксимационные алгоритмы для NP-трудных задач.</li> <li>8. Параллельные и распределенные вычислительные алгоритмы.</li> <li>9. Структуры данных для эффективной работы с большими объемами информации.</li> <li>10. Алгоритмы машинного обучения и их связь с теорией обработки данных.</li> </ol>
---	---

### Шкала оценивания

Шкала оценивания	Критерии оценивания
«Отлично»	Обучающийся глубоко и содержательно раскрывает тему доклада, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер.
«Хорошо»	Обучающийся в целом раскрывает тему доклада, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера.
«Удовлетворительно»	Обучающийся в целом раскрывает тему доклада и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не владеет выбранной темой

### 6.2.3. Оценочные средства остаточных знаний (тест)

#### ПК-3.

1. Что такое структура данных?

- 1) Алгоритм решения задачи
- 2) Организация и способ хранения данных в памяти
- 3) Программа на языке Python
- 4) Файл с данными

2. Какой из перечисленных является линейной структурой данных?

- 1) Дерево
- 2) Граф
- 3) Связный список
- 4) Хеш-таблица

3. Что из перечисленного является примером алгоритма сортировки?

- 1) Быстрая сортировка
- 2) Поиск в глубину

- 3) Обход дерева
- 4) Поиск минимального элемента

4. Какая структура данных использует принцип FIFO?

- 1) Стек
- 2) Очередь
- 3) Хеш-таблица
- 4) Граф

5. Что такое стек?

- 1) Структура данных, работающая по принципу "последним пришёл — первым вышел"
- 2) Структура данных с доступом к элементам по ключу
- 3) Структура данных для хранения дерева
- 4) Система файлов

6. Какой алгоритм поиска работает за логарифмическое время на отсортированном массиве?

- 1) Линейный поиск
- 2) Двоичный поиск
- 3) Поиск в ширину
- 4) Быстрая сортировка

7. Что такое связный список?

- 1) Массив с элементами одинакового типа
- 2) Набор элементов, связанных указателями
- 3) Структура данных для хранения файлов
- 4) Функция сортировки

8. Какая сложность алгоритма быстрой сортировки в худшем случае?

- 1)  $O(n)$
- 2)  $O(n \log n)$
- 3)  $O(n^2)$
- 4)  $O(\log n)$

9. Что такое хеш-таблица?

- 1) Таблица с ключами и значениями, обеспечивающая быстрый доступ по ключу
- 2) Линейная структура данных
- 3) Структура для хранения графа
- 4) Алгоритм сортировки

10. Что из перечисленного является алгоритмом обхода графа?

- 1) Быстрая сортировка
- 2) Поиск в глубину

- 3) Двоичный поиск
- 4) Хеширование

11. Какая структура данных лучше всего подходит для реализации рекурсии?

- 1) Очередь
- 2) Стек
- 3) Хеш-таблица
- 4) Массив

12. Что такое двоичное дерево?

- 1) Дерево, в котором каждый узел имеет не более двух потомков
- 2) Структура данных без связей
- 3) Таблица с ключами
- 4) Алгоритм сортировки

13. Какая структура данных наиболее подходит для реализации очереди с приоритетом?

- 1) Массив
- 2) Двоичная куча (heap)
- 3) Стек
- 4) Связный список

14. Что такое алгоритм поиска в ширину (BFS)?

- 1) Обход графа по уровням
- 2) Обход графа в глубину
- 3) Алгоритм сортировки
- 4) Метод поиска в массиве

15. Какая структура данных используется для хранения элементов в хеш-таблице при коллизиях?

- 1) Стек
- 2) Связный список
- 3) Дерево
- 4) Массив

16. Что означает сложность алгоритма  $O(n \log n)$ ?

- 1) Время выполнения растёт линейно с размером входных данных
- 2) Время выполнения растёт быстрее, чем линейно, но медленнее, чем квадратично
- 3) Время выполнения постоянно
- 4) Время выполнения экспоненциально

17. Что из перечисленного не является алгоритмом сортировки?

- 1) Сортировка вставками

- 2) Быстрая сортировка
- 3) Поиск минимального элемента
- 4) Сортировка слиянием

18. Что такое двоичный поиск?

- 1) Поиск элемента в неотсортированном массиве
- 2) Разбиение массива пополам при поиске элемента в отсортированном массиве
- 3) Обход дерева
- 4) Сортировка массива

19. Какая структура данных используется для реализации рекурсивных вызовов?

- 1) Стек вызовов
- 2) Очередь
- 3) Хеш-таблица
- 4) Граф

20. Что из перечисленного относится к алгоритмам жадного типа?

- 1) Алгоритм Дейкстры
- 2) Поиск в глубину
- 3) Быстрая сортировка
- 4) Двоичный поиск

21. Что такое циклический связный список?

- 1) Список, у которого последний элемент ссылается на первый
- 2) Список с одним элементом
- 3) Список с двумя головами
- 4) Несвязный список

22. Какой из алгоритмов является жадным?

- 1) Алгоритм Крускала
- 2) Двоичный поиск
- 3) Сортировка выбором
- 4) Поиск в ширину

23. Что из перечисленного не является линейной структурой данных?

- 1) Массив
- 2) Стек
- 3) Дерево
- 4) Очередь

24. Что такое динамическое программирование?

1) Метод решения задач путем разбиения на подзадачи и запоминания результатов

- 2) Алгоритм сортировки
- 3) Алгоритм поиска в глубину
- 4) Структура данных

25. Какая структура данных подходит для реализации LRU-кэша?

- 1) Очередь с приоритетом
- 2) Связный список + хеш-таблица
- 3) Массив
- 4) Дерево

26. Что такое обход в глубину (DFS)?

- 1) Алгоритм обхода графа, исследующий как можно дальше каждый путь
- 2) Обход графа по уровням
- 3) Алгоритм сортировки
- 4) Метод поиска минимального элемента

27. Что такое хеш-функция?

- 1) Функция для сортировки данных
- 2) Функция, преобразующая ключ в индекс массива
- 3) Метод поиска в дереве
- 4) Алгоритм обхода графа

28. Какой алгоритм используется для поиска кратчайшего пути в графе с неотрицательными весами ребер?

- 1) Алгоритм Дейкстры
- 2) Быстрая сортировка
- 3) Алгоритм Беллмана-Форда
- 4) Поиск в ширину

29. Что такое сбалансированное дерево?

1) Дерево, в котором высоты левого и правого поддеревьев отличаются не более чем на 1

- 2) Дерево без потомков
- 3) Несвязное дерево
- 4) Структура данных для хранения таблиц

30. Что такое очередь с приоритетом?

1) Очередь, где элементы извлекаются по приоритету, а не по времени вставки

- 2) Простая очередь FIFO
- 3) Стек
- 4) Х е ш - т а б л и ц а

31. Что означает сложность алгоритма  $O(1)$ ?

- 1) Время выполнения не зависит от размера входных данных
- 2) Время выполнения растёт линейно
- 3) Время выполнения экспоненциально
- 4) Время выполнения квадратично

32. Что такое граф?

- 1) Множество узлов и ребер, соединяющих их
- 2) Структура данных для хранения строк
- 3) Алгоритм сортировки
- 4) Таблица с ключами

33. Что такое алгоритм жадного типа?

- 1) Алгоритм, принимающий на каждом шаге локально оптимальное решение
- 2) Алгоритм обхода графа
- 3) Алгоритм сортировки
- 4) Алгоритм поиска

34. Что из перечисленного не относится к типам структур данных?

- 1) Линейные
- 2) Иерархические
- 3) Графовые
- 4) Математические

35. Что такое алгоритм сортировки слиянием?

- 1) Алгоритм, разделяющий массив на две части и сортирующий их рекурсивно
- 2) Алгоритм, сортирующий массив вставками
- 3) Поиск максимального элемента
- 4) Поиск элемента в массиве

36. Что такое элемент данных?

- 1) Минимальная единица информации в структуре данных
- 2) Массив
- 3) Алгоритм
- 4) Таблица

37. Что такое двоичный поиск?

- 1) Метод поиска в неотсортированном массиве
- 2) Метод поиска с делением массива пополам
- 3) Метод сортировки
- 4) Обход графа

38. Что такое алгоритм Крускала?

- 1) Алгоритм поиска минимального остовного дерева
- 2) Алгоритм обхода дерева
- 3) Алгоритм поиска в ширину
- 4) Алгоритм сортировки

39. Что такое сложность алгоритма?

- 1) Количество строк кода
- 2) Количество операций, зависящее от размера входных данных
- 3) Количество памяти
- 4) Количество функций

40. Что из перечисленного является примером неравномерной структуры данных?

- 1) Массив
- 2) Связный список
- 3) Граф
- 4) Очередь

41. Что такое рекурсия?

- 1) Функция, вызывающая сама себя
- 2) Цикл в программе
- 3) Структура данных
- 4) Алгоритм сортировки

42. Что такое алгоритм Беллмана-Форда?

- 1) Алгоритм поиска кратчайших путей с возможными отрицательными весами ребер
- 2) Алгоритм обхода графа
- 3) Алгоритм сортировки
- 4) Поиск максимального элемента

43. Что такое граф с весами?

- 1) Граф, ребра которого имеют числовые значения
- 2) Граф без ребер
- 3) Линейная структура данных
- 4) Стек

44. Что из перечисленного относится к динамическим структурам данных?

- 1) Массив фиксированной длины
- 2) Связный список
- 3) Таблица
- 4) Граф без ребер

45. Что такое жадный алгоритм?

- 1) Алгоритм, принимающий локально оптимальное решение на каждом шаге
- 2) Алгоритм, работающий за константное время
- 3) Алгоритм сортировки
- 4) Алгоритм поиска

**Ключ к тесту:**

1.1	2.3	3.1	4.2	5.1	6.2	7.2	8.3	9.1
10.2	11.2	12.1	13.2	14.1	15.2	16.2	17.3	18.2
19.1	20.1	21.1	22.1	23.3	24.1	25.2	26.1	27.2
28.1	29.1	30.1	31.1	32.1	33.1	34.4	35.1	36.1
37.2	38.1	39.2	40.3	41.1	42.1	43.1	44.2	45.1

### Шкала оценивания результатов тестирования

% верных решений (ответов)	Шкала оценивания
85 - 100	отлично
70 - 84	хорошо
50- 69	удовлетворительно
0 - 49	неудовлетворительно

#### 6.2.4. Примеры задач при разборе конкретных ситуаций

*Тема 1. Введение в дисциплину. Типы и структуры данных. Основные понятия и определения*

##### **ПК-3.**

1. Составить таблицу основных абстрактных типов данных (АТД), указав их свойства и примеры реализации в языках программирования.
2. Определить различия между простыми, агрегатными и ссылочными типами данных на примере одного языка программирования.
3. Построить диаграмму взаимосвязи между различными структурами данных (массив, список, стек, очередь, дерево, граф).
4. Проанализировать задачи, в которых использование определённых типов данных позволяет существенно повысить эффективность.

*Тема 2. Типовые алгоритмы обработки данных*

##### **ПК-3.**

1. Реализовать алгоритмы линейного и бинарного поиска; сравнить их эффективность на отсортированных и неотсортированных массивах.
2. Выполнить реализацию алгоритмов сортировки: пузырьковой, быстрой и сортировки слиянием; сравнить их по времени работы на наборах разного размера.
3. Построить диаграмму алгоритма поиска максимального значения в массиве и выполнить трассировку для конкретного примера.
4. Оценить вычислительную сложность различных методов поиска и сортировки в нотации O-большое.

*Тема 3. Статические и динамические агрегатные структуры данных*  
**ПК-3.**

1. Сравнить работу со структурами "массив" и "связанный список": реализовать вставку и удаление элементов, измерить время выполнения.
2. Реализовать кольцевой связанный список и описать его отличие от обычного односвязного и двусвязного списков.
3. Построить структуру "очередь с приоритетом" и продемонстрировать работу базовых операций добавления и удаления.
4. Реализовать структуру "стек" с использованием массива и списка; сравнить поведение при переполнении.

*Тема 4. Структура данных дерево и алгоритмы на деревьях*  
**ПК-3.**

1. Реализовать бинарное дерево поиска (BST) с функциями вставки, поиска и удаления; построить графическое представление дерева.
2. Выполнить обход дерева в ширину и в глубину (in-order, pre-order, post-order); отразить результаты обходов на примере.
3. Реализовать сбалансированное дерево (например, AVL или красно-черное дерево) и объяснить механизм балансировки.
4. Рассмотреть сценарии применения деревьев в файловых системах и поисковых индексах.

*Тема 5. Структура данных граф и алгоритмы на графах*  
**ПК-3.**

1. Представить граф в виде матрицы смежности и списка смежности; реализовать эти представления в коде.
2. Реализовать алгоритм поиска в глубину (DFS) и поиска в ширину (BFS) для графа; сравнить порядок обхода.
3. Реализовать алгоритм Дейкстры для поиска кратчайшего пути в взвешенном графе; применить его к конкретной задаче маршрутизации.
4. Выполнить реализацию алгоритма поиска компонент связности и визуализировать результат на графе.

*Тема 6. Вычислительные и сложные алгоритмы*  
**ПК-3.**

1. Реализовать жадный алгоритм решения задачи о рюкзаке и сравнить его результат с методом полного перебора.
2. Реализовать алгоритм динамического программирования (например, для задачи о размене монет) и объяснить шаги его работы.
3. Сравнить подходы «разделяй и властвуй» и «динамическое программирование» на конкретном примере (например, сортировка или поиск оптимального пути).
4. Рассчитать асимптотическую сложность ряда алгоритмов и построить граф зависимости времени выполнения от размера входных данных.

**Шкала оценивания**

Шкала оценивания	Критерии оценивания
«Отлично»	обучающийся ясно изложил условие задачи, решение обосновал

«Хорошо»	обучающийся ясно изложил условие задачи, но в обосновании решения имеются сомнения;
«Удовлетворительно»	обучающийся изложил решение задачи, но обосновал его формулировками обыденного мышления;
«Неудовлетворительно»	обучающийся не уяснил условие задачи, решение не обосновал либо не сдал работу на проверку (в случае проведения решения задач в письменной форме).

### 6.2.5. Темы для рефератов

Тема (раздел)	Вопросы
Тема 1. Введение в дисциплину. Типы и структуры данных. Основные понятия и определения	<p>ПК-3</p> <ol style="list-style-type: none"> <li>1. Основные понятия и термины в области структур данных и алгоритмов.</li> <li>2. Классификация типов данных: примитивные и составные типы.</li> <li>3. Структуры данных: определение, назначение и примеры.</li> <li>4. Принципы выбора структуры данных для конкретных задач.</li> <li>5. Понятие абстрактных типов данных (АТД): определение и применение.</li> <li>6. Влияние структуры данных на производительность алгоритмов.</li> <li>7. Роль структур данных в современных языках программирования.</li> <li>8. Применение структур данных в реальных задачах: примеры из практики.</li> <li>9. Сравнительный анализ различных структур данных: массивы, списки, множества.</li> <li>10. Будущее структур данных: новые подходы и технологии.</li> </ol>
Тема 2. Типовые алгоритмы обработки данных	<p>ПК-3</p> <ol style="list-style-type: none"> <li>1. Основные алгоритмы сортировки: пузырьковая, быстрая, сортировка слиянием.</li> <li>2. Алгоритмы поиска: линейный и бинарный поиск, их преимущества и недостатки.</li> <li>3. Алгоритмы обработки строк: поиск подстроки, алгоритмы Кнута-Морриса-Пратта.</li> <li>4. Хэширование: принципы, методы и применение.</li> <li>5. Алгоритмы сжатия данных: принципы и практическое применение.</li> <li>6. Алгоритмы для работы с множествами: объединение, пересечение и разность.</li> <li>7. Алгоритмы обработки графов: поиск в глубину и поиск в ширину.</li> <li>8. Алгоритмы динамического программирования: определение и примеры.</li> <li>9. Применение жадных алгоритмов: принципы и примеры.</li> <li>10. Сравнительный анализ типовых алгоритмов обработки данных.</li> </ol>
Тема 3. Статические и динамические агрегатные структуры данных	<p>ПК-3</p> <ol style="list-style-type: none"> <li>1. Определение статических и динамических структур данных: основные различия.</li> <li>2. Примеры статических агрегатных структур данных: массивы и записи.</li> <li>3. Динамические структуры данных: списки, стек и очередь.</li> <li>4. Применение динамических структур данных в реальных приложениях.</li> <li>5. Статические и динамические структуры данных в контексте</li> </ol>

	<p>оптимизации памяти.</p> <p>6. Алгоритмы работы со статическими и динамическими структурами данных.</p> <p>7. Преимущества и недостатки статических и динамических структур данных.</p> <p>8. Использование комбинированных структур данных: примеры и применение.</p> <p>9. Влияние выбора структуры данных на производительность алгоритмов.</p> <p>10. Будущее агрегатных структур данных: новые подходы и тенденции.</p>
Тема 4. Структура данных дерево и алгоритмы на деревьях	<p>ПК-3</p> <p>1. Определение и классификация деревьев: бинарные деревья, AVL-деревья, красно-черные деревья.</p> <p>2. Алгоритмы обхода деревьев: обход в глубину и в ширину.</p> <p>3. Применение деревьев в хранении и поиске данных: примеры.</p> <p>4. Сравнительный анализ различных типов деревьев: преимущества и недостатки.</p> <p>5. Алгоритмы вставки и удаления узлов в деревьях.</p> <p>6. Балансировка деревьев: техники и алгоритмы.</p> <p>7. Использование деревьев в современных базах данных.</p> <p>8. Применение деревьев для представления иерархических данных.</p> <p>9. Алгоритмы на деревьях в области искусственного интеллекта.</p> <p>10. Будущее деревьев в структурировании и обработке данных: тенденции и новшества.</p>
Тема 5. Структура данных граф и алгоритмы на графах	<p>ПК-3</p> <p>1. Определение графов: типы графов, представление и свойства.</p> <p>2. Алгоритмы поиска в графах: поиск в глубину и поиск в ширину.</p> <p>3. Алгоритмы нахождения кратчайшего пути: алгоритм Дейкстры и алгоритм Флойда-Уоршелла.</p> <p>4. Применение графов в социальных сетях и сетевых технологиях.</p> <p>5. Алгоритмы обхода графов: применение и примеры.</p> <p>6. Использование графов для решения задач маршрутизации.</p> <p>7. Применение графов в машинном обучении и анализе данных.</p> <p>8. Графы и их использование в компьютерной графике.</p> <p>9. Алгоритмы нахождения остовного дерева: алгоритм Краскала и алгоритм Прима.</p> <p>10. Будущее графов в обработке данных: новые технологии и подходы.</p>
Тема 6. Вычислительные и сложные алгоритмы	<p>ПК-3</p> <p>1. Понятие вычислительных алгоритмов: определение и примеры.</p> <p>2. Сложные алгоритмы: что это такое и как они используются.</p> <p>3. Алгоритмы, использующие рекурсию: преимущества и недостатки.</p> <p>4. Применение параллельных и распределенных алгоритмов.</p> <p>5. Сложные алгоритмы в контексте обработки больших данных.</p> <p>6. Алгоритмы, использующие методы машинного обучения.</p> <p>7. Алгоритмы криптографии: основы и примеры.</p> <p>8. Использование вычислительных алгоритмов в научных исследованиях.</p> <p>9. Сравнительный анализ вычислительных и сложных алгоритмов.</p> <p>10. Будущее вычислительных алгоритмов: новые тренды и</p>

### Шкала оценивания

Шкала оценивания	Критерии оценивания
«Отлично»	Обучающийся глубоко и содержательно раскрывает тему самостоятельной работы, не допустив ошибок. Ответ носит развернутый и исчерпывающий характер.
«Хорошо»	Обучающийся в целом раскрывает тему самостоятельной работы, однако ответ хотя бы на один из них не носит развернутого и исчерпывающего характера.
«Удовлетворительно»	Обучающийся в целом раскрывает тему самостоятельной работы и допускает ряд неточностей, фрагментарно раскрывает содержание теоретических вопросов или их раскрывает содержательно, но допуская значительные неточности.
«Неудовлетворительно»	Обучающийся не владеет выбранной темой самостоятельной работы

#### 6.2.6. Индивидуальные задания для курсовой работы (проекта)

Обучающимся предоставляется право выбора темы курсовой работы в соответствии с разработанным перечнем, или обучающийся может предложить свою тему с обоснованием ее актуальности и целесообразности исследования. Во всех случаях тема курсовой работы должна быть согласована с научным руководителем.

Методические указания к курсовой работе по дисциплине «Структуры и алгоритмы обработки данных» к рабочей программе дисциплины прилагаются.

#### Тематика курсовых работ

1. Реализовать структуру данных и алгоритм для хеш-таблицы с методом цепочек.
2. Реализовать структуру данных и алгоритм для хеш-таблицы с открытой адресацией.
3. Реализовать структуру данных и алгоритм для очереди на основе циклического буфера.
4. Реализовать структуру данных и алгоритм для стека с динамическим расширением.
5. Реализовать структуру данных и алгоритм для двусвязного списка с произвольным доступом.
6. Реализовать структуру данных и алгоритм для бинарной кучи.
7. Реализовать структуру данных и алгоритм для очереди с приоритетом на основе кучи.
8. Реализовать структуру данных и алгоритм для декартова дерева.
9. Реализовать структуру данных и алгоритм для красно-чёрного дерева.
10. Реализовать структуру данных и алгоритм для AVL-дерева с балансировкой.
11. Реализовать структуру данных и алгоритм для B-дерева.

12. Реализовать структуру данных и алгоритм для префиксного дерева.
13. Реализовать структуру данных и алгоритм для суффиксного дерева.
14. Реализовать структуру данных и алгоритм для разреженной матрицы.
15. Реализовать структуру данных и алгоритм для неориентированного графа на матрице смежности.
16. Реализовать структуру данных и алгоритм для ориентированного графа на списках смежности.
17. Реализовать структуру данных и алгоритм для обхода графа в ширину.
18. Реализовать структуру данных и алгоритм для обхода графа в глубину.
19. Реализовать структуру данных и алгоритм для поиска кратчайшего пути в графе.
20. Реализовать структуру данных и алгоритм для поиска минимального остовного дерева.
21. Реализовать структуру данных и алгоритм для топологической сортировки вершин графа.
22. Реализовать структуру данных и алгоритм для поиска эйлера цикла в графе.
23. Реализовать структуру данных и алгоритм для поиска гамильтонова цикла в графе.
24. Реализовать структуру данных и алгоритм для поиска компонент сильной связности графа.
25. Реализовать структуру данных и алгоритм для поиска максимального потока в сети.
26. Реализовать структуру данных и алгоритм для быстрой сортировки.
27. Реализовать структуру данных и алгоритм для сортировки слиянием.
28. Реализовать структуру данных и алгоритм для пирамидальной сортировки.
29. Реализовать структуру данных и алгоритм для сортировки подсчётом.
30. Реализовать структуру данных и алгоритм для поразрядной сортировки.
31. Реализовать структуру данных и алгоритм для сортировки Шелла.
32. Реализовать структуру данных и алгоритм для внешней сортировки слиянием.
33. Реализовать структуру данных и алгоритм для двоичного поиска в отсортированном массиве.
34. Реализовать структуру данных и алгоритм для интерполяционного поиска.
35. Реализовать структуру данных и алгоритм для поиска подстроки.
36. Реализовать структуру данных и алгоритм для сжатия данных методом Хаффмана.

37. Реализовать структуру данных и алгоритм для сжатия данных методом LZW.
38. Реализовать структуру данных и алгоритм для кодирования данных алгоритмом RLE.
39. Реализовать структуру данных и алгоритм для вычисления значения многочлена.
40. Реализовать структуру данных и алгоритм для быстрого возведения в степень.
41. Реализовать структуру данных и алгоритм для умножения матриц.
42. Реализовать структуру данных и алгоритм для транспонирования разреженной матрицы.
43. Реализовать структуру данных и алгоритм для нахождения наибольшей общей подпоследовательности.
44. Реализовать структуру данных и алгоритм для нахождения наибольшей общей подстроки.
45. Реализовать структуру данных и алгоритм для вычисления расстояния Левенштейна.
46. Реализовать структуру данных и алгоритм для проверки сбалансированности скобок в выражении.
47. Реализовать структуру данных и алгоритм для преобразования инфиксной записи в постфиксную.
48. Реализовать структуру данных и алгоритм для вычисления выражения в постфиксной записи.
49. Реализовать структуру данных и алгоритм для генерации всех перестановок множества.
50. Реализовать структуру данных и алгоритм для генерации всех сочетаний множества.
51. Реализовать структуру данных и алгоритм для генерации всех подмножеств множества.
52. Реализовать структуру данных и алгоритм для задачи о восьми ферзях.
53. Реализовать структуру данных и алгоритм для задачи о ходе коня на шахматной доске.
54. Реализовать структуру данных и алгоритм для задачи коммивояжёра.
55. Реализовать структуру данных и алгоритм для задачи о рюкзаке.
56. Реализовать структуру данных и алгоритм для поиска медианы в неотсортированном массиве.
57. Реализовать структуру данных и алгоритм для поиска  $k$ -го наименьшего элемента.
58. Реализовать структуру данных и алгоритм для слияния нескольких отсортированных списков.
59. Реализовать структуру данных и алгоритм для проверки принадлежности точки многоугольнику.
60. Реализовать структуру данных и алгоритм для построения выпуклой оболочки.

61. Реализовать структуру данных и алгоритм для вычисления площади многоугольника.
62. Реализовать структуру данных и алгоритм для поиска пересечения двух отрезков.
63. Реализовать структуру данных и алгоритм для триангуляции простого многоугольника.
64. Реализовать структуру данных и алгоритм для поиска ближайшей пары точек на плоскости.
65. Реализовать структуру данных и алгоритм для построения диаграммы Вороного.
66. Реализовать структуру данных и алгоритм для поиска кратчайшего пути в лабиринте.
67. Реализовать структуру данных и алгоритм для генерации лабиринта.
68. Реализовать структуру данных и алгоритм для работы с системой непересекающихся множеств.
69. Реализовать структуру данных и алгоритм для проверки графа на двудольность.
70. Реализовать структуру данных и алгоритм для поиска мостов в графе.
71. Реализовать структуру данных и алгоритм для поиска точек сочленения в графе.
72. Реализовать структуру данных и алгоритм для вычисления числа Каталана.
73. Реализовать структуру данных и алгоритм для быстрого преобразования Фурье.
74. Реализовать структуру данных и алгоритм для поиска подстроки алгоритмом Кнута-Морриса-Пратта.
75. Реализовать структуру данных и алгоритм для поиска подстроки алгоритмом Бойера-Мура.
76. Реализовать структуру данных и алгоритм для поиска минимального остовного дерева алгоритмом Прима.
77. Реализовать структуру данных и алгоритм для поиска минимального остовного дерева алгоритмом Краскала.
78. Реализовать структуру данных и алгоритм для поиска кратчайшего пути алгоритмом Дейкстры.
79. Реализовать структуру данных и алгоритм для поиска кратчайшего пути алгоритмом Флойда-Уоршелла.
80. Реализовать структуру данных и алгоритм для сортировки слиянием без использования дополнительной памяти.

### **Шкала оценивания**

Шкала оценивания	Критерии оценивания
«Отлично»	ставится за курсовую работу, которая характеризуется использованием большого количества новейших литературных источников, глубоким анализом привлеченного материала, творческим подходом к его изложению, знанием закономерностей

	<p>функционирования современных информационных систем, основных понятий, категорий и инструментов в области информатики и вычислительной техники, основных особенностей ведущих школ и направлений в сфере ИТ; умением анализировать и интерпретировать данные отечественной и зарубежной статистики об информационных процессах и явлениях, выявлять тенденции, прогнозировать возможность их развития в будущем, выявлять проблемы технического и алгоритмического характера при анализе конкретных ситуаций, предлагать способы их решения, оценивать риски и возможные технические последствия тех или иных явлений, происходящих в сфере информатики и вычислительной техники. Работа по НИР получает наивысшую оценку в случае одновременного выполнения следующих условий:</p> <p>а) объект исследования описан с предельно широким привлечением источников (как внутренних, так и внешних), на него составлено соответствующее досье, в которое скопированы все использованные материалы;</p> <p>б) самостоятельно и корректно (т.е. в соответствии с реальными фактами) сделаны выводы из анализа досье;</p> <p>в) выявлена взаимосвязь полученных результатов с общетеоретическими проблемами курса микроэкономики.</p> <p>Вынесенные в Приложение материалы могут повысить общую оценку за курсовую работу.</p>
«Хорошо»	<p>ставится за курсовую работу, написанную на достаточно высоком теоретическом уровне, в полной мере раскрывающую содержание темы курсовой работы, с приведенным фактическим материалом, по которому сделаны правильные выводы и обобщения, произведена увязка теории с практикой современной действительности, правильно оформленную работу.</p>
«Удовлетворительно»	<p>ставится за курсовую работу, в которой недостаточно полно освещены узловые вопросы темы, работа написана на базе очень небольшого количества источников, либо на базе устаревших источников.</p>
«Неудовлетворительно»	<p>ставится за курсовую работу, переписанную с одного или нескольких источников. Работа в рамках НИР оценивается неудовлетворительно в случае нарушения требований задания.</p>

### **6.3. ОЦЕНОЧНЫЕ СРЕДСТВА ПРОМЕЖУТОЧНОГО КОНТРОЛЯ**

**Вопросы для подготовки к промежуточной аттестации по итогам освоения дисциплины Структуры и алгоритмы обработки данных:**

#### **ПК-3.**

1. Основные понятия структуры данных
2. Архитектура баз данных
3. Типы данных.
4. Концепция типа данных.
5. Двоичные целые числа со знаком и без знака
6. Кодирование символов.
7. Кодовые страницы
8. Типовые алгоритмы обработки данных
9. Прохождение пользовательского запроса

10. Пользователи банков данных
11. Файл. Доступ к данным файлов.
12. Структура файловой системы ОС. Сортировка файлов.
13. Алгоритмы поиска в файле. Использование индексных файлов.
14. В-дерево и поиск данных в файле.
15. Классификация моделей данных
16. Структура данных дерево
17. Алгоритмы на деревьях
18. Статические агрегатные структуры: массивы, матрицы, записи, таблицы
19. Алгоритмы простых сортировок: пузырьковая, вставками, методом выбора
20. Параметры алгоритмов сортировки.
21. Время работы простых алгоритмов сортировки
22. Алгоритмы устойчивых и неустойчивых сортировок
23. Структура данных граф
24. Алгоритмы на графах
25. Направленные и ненаправленные графы.
26. Традиционные операции над множествами: объединение, пересечение, разность и декартово произведение.
27. Специальные реляционные операции: выборка, проекция, соединение, деление.
28. O-нотация.
29. Порядок сложности алгоритмов
30. Шейкерная сортировка
31. Сортировка Шелла
32. Последовательный и бинарный поиск
33. Сортировка массива записей по ключу
34. Рекурсия.
35. Быстрая сортировка
36. Динамические структуры данных: линейный односвязный список
37. Динамические структуры данных: линейный двусвязный список
38. Операции с односвязным списком: добавление узла в начало списка, после заданного, перед заданным, в конец списка
39. Поиск узла в списке
40. Удаление узла в списке
41. Операции с двусвязным списком: добавление узла в начало списка, добавление узла в конец списка
42. Добавление узла после заданного, поиск узла в списке, удаление узла
43. Динамические структуры данных: стеки и очереди.
44. Операции с динамическими структурами данных
45. Реализация стека на массиве и списках
46. Реализация очереди на массиве и списках
47. DDL операторы языка SQL
48. DML операторы языка SQL

49. DQL операторы языка SQL
50. Алгоритмы сжатия данных. Характеристики. Словарные и статистические методы сжатия данных. сжатия информации.
51. Алгоритм сжатия, применяемый в архиваторе Zip.
52. Полный перебор. Метод динамического программирования.
53. Метод ветвей и границ.
54. Эвристический поиск. Поиск по образцу.
55. Алгоритмы аппроксимации числовых функций.
56. Стандартная и каноническая формы задач линейного программирования.
57. Преобразование задач линейного программирования в стандартную (каноническую) форму.
58. Разрешимость задачи линейного программирования.
59. Задача целочисленного линейного программирования (ЦЛП).
60. Метод ветвей и границ для решения задач ЦЛП.
61. Представление графов в памяти: списки смежности и матрицы смежности
62. Алгоритм поиска в глубину (DFS) и его применение
63. Алгоритм поиска в ширину (BFS) и его применение
64. Алгоритм Дейкстры для поиска кратчайшего пути
65. Алгоритм Беллмана-Форда: особенности и применение
66. Алгоритм Флойда–Уоршелла: нахождение всех кратчайших путей
67. Алгоритмы построения минимального остовного дерева (Краскал, Прим)
68. Хеш-таблицы: структура, применение, разрешение коллизий
69. Двоичное дерево поиска (BST): вставка, удаление, поиск
70. Самобалансирующиеся деревья (AVL-деревья, красно-черные деревья)
71. Куча (Heap) и реализация алгоритма пирамидальной сортировки (HeapSort)
72. Префиксные деревья (Trie) и их применение в поиске
73. Aho–Corasick: алгоритм поиска подстрок
74. Обзор современных алгоритмов поиска данных в больших объемах
75. Выбор структуры данных под задачу: критерии и примеры

#### **6.4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

Основной целью проведения промежуточной аттестации является определение степени достижения целей по учебной дисциплине или ее разделам. Осуществляется это проверкой и оценкой уровня теоретических знаний, полученных обучающимися, умения применять их в решении практических задач, степени овладения обучающимися практическими навыками и умениями в объеме требований рабочей программы по дисциплине, а также их умение самостоятельно работать с учебной литературой.

Организация проведения промежуточной аттестации регламентирована «Положением об организации образовательного процесса в федеральном государственном автономном образовательном учреждении «Московский политехнический университет».

#### 6.4.1. Показатели оценивания компетенций на различных этапах их формирования, достижение обучающимися планируемых результатов обучения по дисциплине

<b>Код и наименование компетенции ПК-3. Способен разрабатывать компоненты системных программных продуктов</b>				
<b>Этап (уровень)</b>	<b>Критерии оценивания</b>			
	<b>неудовлетворительно</b>	<b>удовлетворительно</b>	<b>хорошо</b>	<b>отлично</b>
<b>знать</b>	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие следующих знаний: назначение, организацию, принципы функционирования, последовательность и этапы разработки системных, инструментальных и прикладных программ, программных комплексов и систем.	Обучающийся демонстрирует неполное соответствие следующих знаний: назначение, организацию, принципы функционирования, последовательность и этапы разработки системных, инструментальных и прикладных программ, программных комплексов и систем.	Обучающийся демонстрирует частичное соответствие следующих знаний: назначение, организацию, принципы функционирования, последовательность и этапы разработки системных, инструментальных и прикладных программ, программных комплексов и систем.	Обучающийся демонстрирует полное соответствие следующих знаний: назначение, организацию, принципы функционирования, последовательность и этапы разработки системных, инструментальных и прикладных программ, программных комплексов и систем.
<b>уметь</b>	Обучающийся не умеет или в недостаточной степени умеет выполнять: применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных.	Обучающийся демонстрирует неполное соответствие следующих умений: применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных.	Обучающийся демонстрирует частичное соответствие следующих умений: применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных.	Обучающийся демонстрирует полное соответствие следующих умений: применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных.

<b>владеть</b>	Обучающийся не владеет или в недостаточной степени владеет: методами и средствами разработки компонентов аппаратно-программных комплексов и баз данных.	Обучающийся владеет в неполном и проявляет недостаточность владения: методами и средствами разработки компонентов аппаратно-программных комплексов и баз данных.	Обучающимся допускаются незначительные ошибки, неточности, затруднения, частично владеет: методами и средствами разработки компонентов аппаратно-программных комплексов и баз данных.	Обучающийся свободно применяет полученные навыки, в полном объеме владеет: методами и средствами разработки компонентов аппаратно-программных комплексов и баз данных.
----------------	---	--	---	--

#### 6.4.2. Методика оценивания результатов промежуточной аттестации

Показателями оценивания компетенций на этапе промежуточной аттестации по дисциплине «Структуры и алгоритмы обработки данных» являются результаты обучения по дисциплине.

#### Оценочный лист результатов обучения по дисциплине

Код компетенции	Знания	Умения	Навыки	Уровень сформированности и компетенции на данном этапе / оценка
ПК-3. Способен разрабатывать компоненты системных программных продуктов	на уровне знаний: знать методы и средства разработки компонентов аппаратно-программных комплексов и баз данных; знать назначение, организацию, принципы функционирования, последовательность и этапы разработки системных, инструментальных и прикладных программ, программных комплексов и систем; знать стандарты, методические и нормативные материалы, определяющие	на уровне умений: уметь применять современные инструментальные средства и технологии программирования при разработке компонентов аппаратно-программных комплексов и баз данных; уметь применять современные программно-методические комплексы автоматизированного проектирования объектов профессиональной деятельности.	на уровне навыков: методами и средствами разработки компонентов аппаратно-программных комплексов и баз данных; методами организации процесса разработки компонентов аппаратно-программных комплексов и баз данных.	

	проектирование и разработку компонентов аппаратно-программных комплексов и баз данных; знать модели, методы и формы организации процесса разработки компонентов аппаратно-программных комплексов и баз данных; знать методы и средства обеспечения информационной безопасности разрабатываемых компонентов АПК			
Оценка по дисциплине (среднее арифметическое)				

Оценка по дисциплине зависит от уровня сформированности компетенций, закрепленных за дисциплиной, и представляет собой среднее арифметическое от выставленных оценок по отдельным результатам обучения (знания, умения, навыки).

Оценка «отлично» выставляется, если среднее арифметическое находится в интервале от 4,5 до 5,0.

Оценка «хорошо» выставляется, если среднее арифметическое находится в интервале от 3,5 до 4,4.

Оценка «удовлетворительно» выставляется, если среднее арифметическое находится в интервале от 2,5 до 3,4.

Оценка «неудовлетворительно» выставляется, если среднее арифметическое находится в интервале от 0 до 2,4.

Промежуточная аттестация обучающихся в форме экзамена проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по дисциплине «Структуры и алгоритмы обработки данных», при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине проводится преподавателем, ведущим занятия по дисциплине методом экспертной оценки. По итогам промежуточной аттестации по дисциплине выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

Шкала оценивания	Описание
Отлично	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Хорошо	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует неполное, правильное соответствие знаний, умений, навыков приведенным в таблицах показателей, либо если при этом были допущены 2-3 несущественные ошибки.
Удовлетворительно	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, в котором освещена основная, наиболее важная часть материала, но при этом допущена одна значительная ошибка или неточность.
Неудовлетворительно	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

## 7. Электронная информационно-образовательная среда

Каждый обучающийся в течение всего периода обучения обеспечивается индивидуальным неограниченным доступом к электронной информационно-образовательной среде Чебоксарского института (филиала) Московского политехнического университета из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети «Интернет» (далее – сеть «Интернет»), как на территории филиала, так и вне ее.

Электронная информационно-образовательная среда – совокупность информационных и телекоммуникационных технологий, соответствующих технологических средств, обеспечивающих освоение обучающимися образовательных программ в полном объеме независимо от места нахождения обучающихся.

Электронная информационно-образовательная среда обеспечивает:

а) доступ к учебным планам, рабочим программам дисциплин (модулей), практик, электронным учебным изданиям и электронным образовательным ресурсам, указанным в рабочих программах дисциплин (модулей), практик;

б) формирование электронного портфолио обучающегося, в том числе сохранение его работ и оценок за эти работы;

в) фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения программы бакалавриата;

г) проведение учебных занятий, процедур оценки результатов обучения, реализация которых предусмотрена с применением электронного обучения, дистанционных образовательных технологий;

д) взаимодействие между участниками образовательного процесса, в том числе синхронное и (или) асинхронное взаимодействия посредством сети «Интернет».

Функционирование электронной информационно-образовательной среды обеспечивается соответствующими средствами информационно-коммуникационных технологий и квалификацией работников, ее использующих и поддерживающих.

Функционирование электронной информационно-образовательной среды соответствует законодательству Российской Федерации.

Основными составляющими ЭИОС филиала являются:

а) сайт института в сети Интернет, расположенный по адресу [www.polytech21.ru](http://www.polytech21.ru), <https://chebpolytech.ru/> который обеспечивает:

- доступ обучающихся к учебным планам, рабочим программам дисциплин, практик, к изданиям электронных библиотечных систем, электронным информационным и образовательным ресурсам, указанных в рабочих программах (разделы сайта «Сведения об образовательной организации»);

- информирование обучающихся обо всех изменениях учебного процесса (новостная лента сайта, лента анонсов);

- взаимодействие между участниками образовательного процесса (подразделы сайта «Задать вопрос директору»);

б) официальные электронные адреса подразделений и сотрудников института с Яндекс-доменом [@polytech21.ru](mailto:@polytech21.ru) (список контактных данных подразделений Филиала размещен на официальном сайте Филиала в разделе «Контакты», списки контактных официальных электронных данных преподавателей размещены в подразделах «Кафедры») обеспечивают взаимодействие между участниками образовательного процесса;

в) личный кабинет обучающегося (портфолио) (вход в личный кабинет размещен на официальном сайте Филиала в разделе «Студенту» подразделе «Электронная информационно-образовательная среда») включает в себя портфолио студента, электронные ведомости, рейтинг студентов и обеспечивает:

- фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения образовательных программ обучающимися,

- формирование электронного портфолио обучающегося, в том числе с сохранение работ обучающегося, рецензий и оценок на эти работы,

г) электронные библиотеки, включающие электронные каталоги, полнотекстовые документы и обеспечивающие доступ к учебно-методическим материалам, выпускным квалификационным работам и т.д.:

Чебоксарского института (филиала) - «ИРБИС»

д) электронно-библиотечные системы (ЭБС), включающие электронный каталог и полнотекстовые документы:

- ЭБС «ЛАНЬ» -<https://e.lanbook.com/>

- Образовательная платформа Юрайт - <https://urait.ru>

- IPR SMART -<https://www.iprbookshop.ru/>

е) платформа цифрового образования Политеха -<https://lms.mospolytech.ru/>

ж) система «Антиплагиат» -<https://www.antiplagiat.ru/>

з) система электронного документооборота DIRECTUM Standard — обеспечивает документооборот между Филиалом и Университетом;

и) система «IC Управление ВУЗом Электронный деканат» (Московский политехнический университет) обеспечивает фиксацию хода образовательного процесса, результатов промежуточной аттестации и результатов освоения образовательных программ обучающимися;

к) система «POLYTECH systems» обеспечивает информационное, документальное автоматизированное сопровождение образовательного процесса;

л) система «Абитуриент» обеспечивает документальное автоматизированное сопровождение работы приемной комиссии.

## **8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины**

### Основная литература:

1. Проектирование информационных систем : учебник и практикум для вузов / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 273 с. — (Высшее образование). — ISBN 978-5-534-20361-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/560485>.

2. Золкин, А. Л. Автоматизация и диспетчеризация систем. Применение языковых средств высокоуровневого программирования : учебник для вузов / А. Л. Золкин, В. Д. Мунистер. — Санкт-Петербург : Лань, 2025. — 164 с. — ISBN 978-5-507-51451-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/450806>. — Режим доступа: для авториз. пользователей.

3. Давыдова, Н. А. Программирование : учебное пособие / Н. А. Давыдова, Е. В. Боровская. — 5-е изд. (эл.). — Москва : Лаборатория знаний, 2025. — 241 с. — ISBN 978-5-93208-831-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/451607>. — Режим доступа: для авториз. пользователей.

### Дополнительная литература:

1. Информационные системы и технологии в экономике и управлении в 2 ч. Часть 1 : учебник для вузов / ответственный редактор В. В. Трофимов. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 375 с. —

(Высшее образование). — ISBN 978-5-534-09090-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/564598>.

2. Информационные системы и технологии в экономике и управлении в 2 ч. Часть 2 : учебник для вузов / ответственный редактор В. В. Трофимов. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 324 с. — (Высшее образование). — ISBN 978-5-534-09092-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/564599>.

#### Периодика:

1. Известия Тульского государственного университета. Технические науки: Научный рецензируемый журнал. <https://tidings.tsu.tula.ru/tidings/index.php?id=technical&lang=ru&year=1>.

- Текст : электронный.

2. Научный периодический журнал «Вестник Южно-Уральского государственного университета. Серия «Вычислительная математика и информатика» : Научный рецензируемый журнал. <https://vestnik.susu.ru/cmi> - Текст : электронный.

3. Научный периодический журнал «Вестник Южно-Уральского государственного университета. Серия «Компьютерные технологии, управление, радиоэлектроника»: Научный рецензируемый журнал. <https://vestnik.susu.ru/ctcr> - Текст : электронный.

### **9. Профессиональные базы данных и информационно-справочные системы**

Профессиональная база данных и информационно-справочные системы	Информация о праве собственности (реквизиты договора)
Университетская информационная система РОССИЯ <a href="https://uisrussia.msu.ru/">https://uisrussia.msu.ru/</a>	Тематическая электронная библиотека и база для прикладных исследований в области экономики, управления, социологии, лингвистики, философии, филологии, международных отношений, права. свободный доступ
научная электронная библиотека Elibrary <a href="http://elibrary.ru/">http://elibrary.ru/</a>	Научная электронная библиотека eLIBRARY.RU - это крупнейший российский информационно-аналитический портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты более 26 млн научных статей и публикаций, в том числе электронные версии более 5600 российских научно-технических журналов, из которых более 4800 журналов в открытом доступе свободный доступ
сайт Института научной информации по общественным наукам РАН. <a href="http://www.inion.ru">http://www.inion.ru</a>	Библиографические базы данных ИНИОН РАН по социальным и гуманитарным наукам ведутся с начала 1980-х годов. Общий объем массивов составляет более 3 млн. 500 тыс. записей (данные на 1 января 2012 г.). Ежегодный прирост — около 100 тыс. записей. В базы данных включаются аннотированные описания книг и статей из журналов и сборников на 140 языках, поступивших в Фундаментальную библиотеку ИНИОН

	РАН. Описания статей и книг в базах данных снабжены шифром хранения и ссылками на полные тексты источников из Научной электронной библиотеки.
Федеральный портал «Российское образование» [Электронный ресурс] – <a href="http://www.edu.ru">http://www.edu.ru</a>	Федеральный портал «Российское образование» – уникальный интернет-ресурс в сфере образования и науки. Ежедневно публикует самые актуальные новости, анонсы событий, информационные материалы для широкого круга читателей. Еженедельно на портале размещаются эксклюзивные материалы, интервью с ведущими специалистами – педагогами, психологами, учеными, репортажи и аналитические статьи. Читатели получают доступ к нормативно-правовой базе сферы образования, они могут пользоваться самыми различными полезными сервисами – такими, как онлайн-тестирование, опросы по актуальным темам и т.д.
Информационные технологии – периодическое научно-техническое издание в области информационных технологий, автоматизированных систем и использования информатики в различных приложениях <a href="http://novtex.ru">novtex.ru</a>	Издательство выпускает теоретические и прикладные научно-технические журналы, обеспечивающие научной, производственной, обзорно-аналитической и образовательной информацией руководящих работников и специалистов промышленных предприятий, научных академических и отраслевых организаций, а также учебных заведений в области приоритетных направлений развития науки и технологий.
Ассоциация инженерного образования России <a href="http://www.ac-raee.ru/">http://www.ac-raee.ru/</a>	Совершенствование образования и инженерной деятельности во всех их проявлениях, относящихся к учебному, научному и технологическому направлениям, включая процессы преподавания, консультирования, исследования, разработки инженерных решений, оказания широкого спектра образовательных услуг, обеспечения связей с общественностью, производством, наукой и интеграции в международное научно-образовательное пространство. свободный доступ

## 10. Программное обеспечение (лицензионное и свободно распространяемое), используемое при осуществлении образовательного процесса

Аудитория	Программное обеспечение	Информация о праве собственности (реквизиты договора, номер лицензии и т.д.)
№ 2196 Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/специалитета/ магистратуры, оснащенная оборудованием и	Windows 7 OLPNLAcdmc	договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License	Сублицензионный договор №977_1049.ЕП/25 от 10.12.2025
	Yandex браузер	Свободное распространяемое программное обеспечение (бессрочная

техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) <u>Кабинет систем управления ООО «НПО «Каскад-ГРУП»</u>		лицензия)
	МТС Линк	Договор №2/2026 (091_168.ЕП/26) от 27.03.2026
	Microsoft Office Standard 2007(Microsoft DreamSpark Premium Electronic Software Delivery Academic (Microsoft Open License	номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
<b>№ 2116</b> Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) <u>Лаборатория «Программного обеспечения и сопровождения компьютерных систем»</u> <u>Кабинет информационных систем и технологий АО «НПК «ЭЛАРА»</u>	Windows 7 OLPNLAcadmс	договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License	Сублицензионный договор №977_1049.ЕП/25 от 10.12.2025
	Microsoft Visual Studio 2019	свободно распространяемое программное обеспечение (бессрочная лицензия)
	КОМПАС-3D v20 и v21	Сублицензионный договор № Нп-22-00044 от 21.03.2022 (бессрочная лицензия)
	PaitNet	свободно распространяемое программное обеспечение (бессрочная лицензия)
	AIMP	отечественное свободно распространяемое программное обеспечение (бессрочная лицензия)
<b>№ 1126</b> Помещение для самостоятельной работы обучающихся	Kaspersky Endpoint Security для бизнеса – Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License	Сублицензионный договор №977_1049.ЕП/25 от 10.12.2025
	Windows 7 OLPNLAcadmс	договор №Д03 от 30.05.2012) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	AdobeReader	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Гарант- справочно-правовая система	Договор №С-002-2025 от 09.01.2025
	Yandex браузер	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Microsoft Office Standard 2007(Microsoft DreamSpark Premium Electronic Software Delivery Academic (Microsoft Open License	номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	МТС Линк	Договор №2/2026 (091_168.ЕП/26) от 27.03.2026
AIMP	отечественное свободно распространяемое программное обеспечение (бессрочная лицензия)	
<b>№ 103а</b> Помещение для самостоятельной работы обучающихся	Kaspersky Endpoint Security для бизнеса –	Сублицензионный договор №977_1049.ЕП/25 от 10.12.2025

	Расширенный Russian Edition. 150-249 Node 2 year Educational Renewal License	
	MS Windows 10 Pro	договор № 392_469.223.3К/19 от 17.12.19 (бессрочная лицензия)
	AdobeReader	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Гарант- справочно-правовая система	Договор №С-002-2025 от 09.01.2025
	Yandex браузер	свободно распространяемое программное обеспечение (бессрочная лицензия)
	Microsoft Office Standard 2007(Microsoft DreamSpark Premium Electronic Software Delivery Academic (Microsoft Open License	номер лицензии-42661846 от 30.08.2007) с допсоглашениями от 29.04.14 и 01.09.16 (бессрочная лицензия)
	AIMP	отечественное свободно распространяемое программное обеспечение (бессрочная лицензия)

## 11. Материально-техническое обеспечение дисциплины

Тип и номер помещения	Перечень основного оборудования и технических средств обучения
Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/ специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Кабинет систем управления ООО «НПО «Каскад-ГРУП» № 2196 (г. Чебоксары, ул. К.Маркса, 60)	<u>Оборудование:</u> комплект мебели для учебного процесса; доска учебная; стенды <u>Технические средства обучения:</u> мультимедийное оборудование (проектор, экран)
Учебная аудитория для проведения учебных занятий всех видов, предусмотренных программой среднего профессионального образования/бакалавриата/ специалитета/ магистратуры, оснащенная оборудованием и техническими средствами обучения, состав которых определяется в рабочих программах дисциплин (модулей) Лаборатория «Программного обеспечения и сопровождения компьютерных систем» Кабинет информационных систем и технологий АО «НПК «ЭЛАРА» № 2116 (г. Чебоксары, ул. К.Маркса, 60)	<u>Оборудование:</u> комплект мебели для учебного процесса; доска учебная; стенды, автоматизированные рабочие места на 15 обучающихся, автоматизированное рабочее место преподавателя, <u>Технические средства обучения:</u> компьютерная техника; мультимедийное оборудование (проектор, экран), маркерная доска, программное обеспечение общего и профессионального назначения
Помещение для самостоятельной работы обучающихся № 1126 (г. Чебоксары, ул. К.Маркса, 60)	<u>Оборудование:</u> комплект мебели для учебного процесса; <u>Технические средства обучения:</u> компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную

	информационно-образовательную среду Филиала
Помещение для самостоятельной работы обучающихся № 103а (г. Чебоксары, ул. К.Маркса, 54)	<u>Оборудование:</u> комплект мебели для учебного процесса; <u>Технические средства обучения:</u> компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Филиала

## 12. Методические указания для обучающегося по освоению дисциплины

### *Методические указания для занятий лекционного типа*

В ходе лекционных занятий обучающемуся необходимо вести конспектирование учебного материала, обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации.

Необходимо задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций. Целесообразно дорабатывать свой конспект лекции, делая в нем соответствующие записи из основной и дополнительной литературы, рекомендованной преподавателем и предусмотренной учебной программой дисциплины.

### *Методические указания для занятий лабораторного типа.*

Выполнению лабораторных работ предшествует проверка знаний студентов – их теоретической готовности к выполнению задания. Проверка знаний проводится в форме, которую определяет преподаватель дисциплины (тестирование, опрос).

При проведении лабораторных занятий выделяют следующие разделы:

- общие положения (перечень лабораторных или практических занятий);
- общие требования к выполнению работ;
- инструкция по каждой работе;
- справочные материалы и т. д.

Лабораторные занятия позволяют развивать у обучающегося творческое теоретическое мышление, умение самостоятельно изучать литературу, анализировать практику; учат четко формулировать мысль, вести дискуссию, то есть имеют исключительно важное значение в развитии самостоятельного мышления.

Готовясь к докладу или выступлению в рамках интерактивной формы, при необходимости, следует обратиться за помощью к преподавателю.

### *Методические указания к самостоятельной работе.*

Самостоятельная работа обучающегося является основным средством овладения учебным материалом вовремя, свободное от обязательных учебных занятий. Самостоятельная работа обучающегося над усвоением учебного

материала по учебной дисциплине может выполняться в библиотеке университета, учебных кабинетах, компьютерных классах, а также в домашних условиях. Содержание и количество самостоятельной работы обучающегося определяется учебной программой дисциплины, методическими материалами, практическими заданиями и указаниями преподавателя.

***Самостоятельная работа в аудиторное время может включать:***

- 1) конспектирование лекций;
- 2) выполнение контрольных работ;
- 3) решение задач;
- 4) работу со справочной и методической литературой;
- 5) работу с нормативными правовыми актами;
- 6) выступления с докладами, сообщениями на семинарских занятиях;
- 7) защиту выполненных работ;
- 8) участие в оперативном (текущем) опросе по отдельным темам изучаемой дисциплины;
- 9) участие в беседах, деловых (ролевых) играх, дискуссиях, круглых столах, конференциях;
- 10) участие в тестировании и др.

***Самостоятельная работа во внеаудиторное время может состоять из:***

- 1) повторения лекционного материала;
- 2) подготовки к практическим занятиям;
- 3) изучения учебной и научной литературы;
- 4) изучения нормативных правовых актов (в т.ч. в электронных базах данных);
- 5) решения задач, и иных практических заданий;
- 6) подготовки к контрольным работам, тестированию и т.д.;
- 7) подготовки к практическим занятиям устных докладов (сообщений);
- 8) выполнения курсовых работ, предусмотренных учебным планом;
- 9) выполнения выпускных квалификационных работ и др.;
- 10) выделения наиболее сложных и проблемных вопросов по изучаемой теме, получение разъяснений и рекомендаций по данным вопросам с преподавателями на консультациях;
- 11) проведения самоконтроля путем ответов на вопросы текущего контроля знаний, решения представленных в учебно-методических материалах кафедры задач, тестов, докладов;
- 12) текущий контроль осуществляется в форме устных, тестовых опросов, докладов;

В случае пропусков занятий, наличия индивидуального графика обучения и для закрепления практических навыков студентам могут быть выданы типовые индивидуальные задания, которые должны быть сданы в установленный преподавателем срок.

### **13. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья**

Обучение по данной дисциплине инвалидов и лиц с ограниченными возможностями здоровья (далее ОВЗ) осуществляется преподавателем с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

Для студентов с нарушениями опорно-двигательной функции и с ОВЗ по слуху предусматривается сопровождение лекций и практических занятий мультимедийными средствами, раздаточным материалом.

Для студентов с ОВЗ по зрению предусматривается применение технических средств усиления остаточного зрения, а также предусмотрена возможность разработки аудиоматериалов.

По данной дисциплине обучение инвалидов и лиц с ограниченными возможностями здоровья может осуществляться как в аудитории, так и с использованием электронной информационно-образовательной среды, образовательного портала и электронной почты.

## ЛИСТ ДОПОЛНЕНИЙ И ИЗМЕНЕНИЙ

### рабочей программы дисциплины

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202\_\_-202\_\_ учебном году на заседании кафедры, протокол № от « \_\_\_\_\_ » \_\_\_\_\_ 202\_\_ г.

Внесены дополнения и изменения \_\_\_\_\_

---

---

---

---

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202\_\_-202\_\_ учебном году на заседании кафедры, протокол № от « \_\_\_\_\_ » \_\_\_\_\_ 202\_\_ г.

Внесены дополнения и изменения \_\_\_\_\_

---

---

---

---

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202\_\_-202\_\_ учебном году на заседании кафедры, протокол № от « \_\_\_\_\_ » \_\_\_\_\_ 202\_\_ г.

Внесены дополнения и изменения \_\_\_\_\_

---

---

---

---

Рабочая программа дисциплины рассмотрена, обсуждена и одобрена для исполнения в 202\_\_-202\_\_ учебном году на заседании кафедры, протокол № от « \_\_\_\_\_ » \_\_\_\_\_ 202\_\_ г.

Внесены дополнения и изменения \_\_\_\_\_

---

---

---

---